EXHIBIT 10

PLATFORM CHOICE BY MOBILE APP DEVELOPERS*

TIMOTHY BRESNAHAN, JOE ORSINI AND PAI-LING YIN

Stanford University

May 29, 2014

For the past two ad a half years, Apple's iOS and Google's Android operating systems have split the market share of smartphone devices and the mobile applications (apps) for those devices. We model the platform choice by mobile app developers and estimate the user preferences and developer's profitability from multihoming (supplying both platforms) implied by the data. We find little difference in preferences across platforms, even when we allow for differences across types of developers and apps. We identify strong incentives for developers of the most popular apps to multihome, making tipping unlikely.

*tbres@stanford.edu, jorsini@stanford.edu, pyin@stanford.edu. This research project is based on data collection and analysis over a wide range of data sources. We are very grateful to a number of research assistants who have worked on those datasets, gathered industry information, and joined us in industry interviews. These include Markus Baldauf, Sean Batir, Robert Burns, Jane Chen, Emanuele Colonnelli, Elizabeth Davis, Sherry Fu, Osama El-Gabalawy, Carlos Garay, Jorge Guzman, Alireza Forouzan Ebrahimi, Tim Jaconette, Nayaranta Jain, Julia Kho, Sigtryggur Kjarttansson, Xing Li, Derek Lief, Sean Mandell, Laura Miron, Jaron Moore, Yulia Muzyrya, Abhishek Nagaraj, Jin Hyung Park, Francis Plaza, Hatim Rahman, Juan Rios, Sam Seyfollahi, Melissa Sussman-Martinez, Masoud Tavazoei, Sylvan Tsai, Julis Vazquez, Sarah Wilson, Joon Young Yoon, Jessica Zhang and Parker Zhao. Special thanks to Jen Brown, Michael Jacobides, and participants at the NBER Economics of Digitization, Wharton Technology and Innovation Conference, and Stanford Social Science and Technology Seminar. We are also very grateful to the many industry participants who have shared their time and expertise with us.

Platforms have often been used in information communications technology industries to successfully harness innovations from diverse sources. Social increasing returns create a positive feedback loop, as the platform with the most users will further attract more developers who attract more users. These indirect network effects tend to tip the market to the platform with the larger market share (Farrell & Klemperer, 2007). Equilibrium platform market structure tends to be concentrated through tipping. The market structure in which two platforms are approximately tied is an equilibrium, but an unstable one; small departures from it by either users or developers tend to lead to a tip away.

These familiar theoretical implications are only partially borne out in the most recent and important example of platforms in information communications technology, mobile applications (apps). Two technically capable and well-funded market participants, BlackBerry and Windows Mobile, have tipped out and are now competitively irrelevant. In many countries, the platform market has tipped to a single important supplier, Google Android. However, the unstable equilibrium appears to hold in the US market, where Apple iOS and Google Android have approximately split the platform market. Measured either by users or by app developers (see Figures I and III), this market has persisted as an approximate tie between the two leading platforms.

To understand this phenomenon, we estimate a new model of user demand for apps and developer choice of platforms. Our user model permits users of each of the two larger platforms to have similar or different demand for apps. Our developer model permits individual developers to enter one or both platforms (i.e., to multihome). We combine hand-constructed data on developers' platform choices and app and developer characteristics with commercial data on app usage. This new dataset plus our model lets us estimate the developers' expected profitability from entering either or both platforms. It also lets us infer user preferences for apps on both platforms.

We find that user preferences vary widely across apps. However, users' preferences for the same app do not vary much at all across platforms. This finding of demand similarity is particularly strong for apps that are successful on either platform; such apps are highly likely to be approximately equally successful on the other platform. Partly because of this demand symmetry, we also find that the decision rule determining app supply to each platform is also similar across platforms, evaluated at the current, approximately evenly split user platform market shares. Finally, we show that these demand and supply results imply that approximately evenly split market shares create incentives for the highest demand apps to multihome.

This multihoming result is the key to explaining the stability of the fragmented platform market equilibrium. Given the fragmented platform market structure, the most important apps multihome. Most of these important apps reach far more consumers on both platforms relative to the threshold level of customers below which they would exit either platform. Thus, the incentives for the most important developers to

FIGURE I: FRAGMENTATION OF USERS. SOURCE: HTTP://WWW.TECH-THOUGHTS.NET/2012/07/GLOBAL-SMARTPHONE-MARKET-SHARE-TRENDS.HTML#.UTDYR_RDTNJ

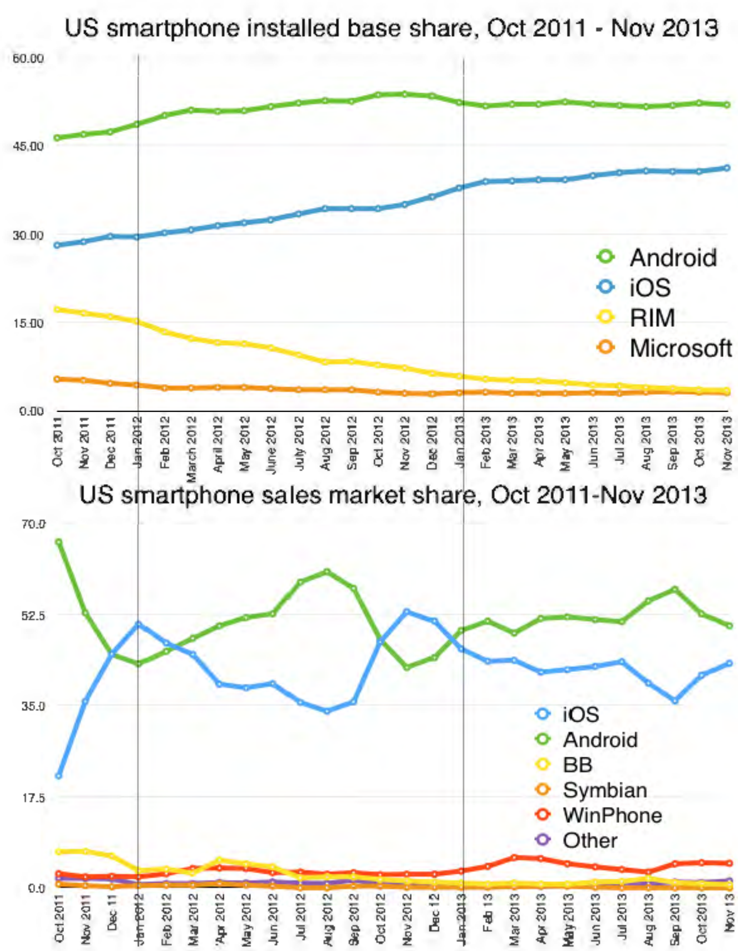


FIGURE II: FRAGMENTATION OF DEVELOPERS IN US APP MARKET. THE LINE OF DOTS REFLECTS IS iOS APPS. THE SPORADIC SQUARES TRACK ANDROID APPS. SOURCE: HTTP://148APPS.BIZ/APP-STORE-METRICS/?MPAGE=APPCOUNT, HTTP://EN.WIKIPEDIA.ORG/WIKI/GOOGLE_PLAY#APPLICATIONS, AND HTTP://WWW.APPBRAIN.COM/STATS/NUMBER-OF-ANDROID-APPS, ALL ACCESSED JANUARY 15, 2014

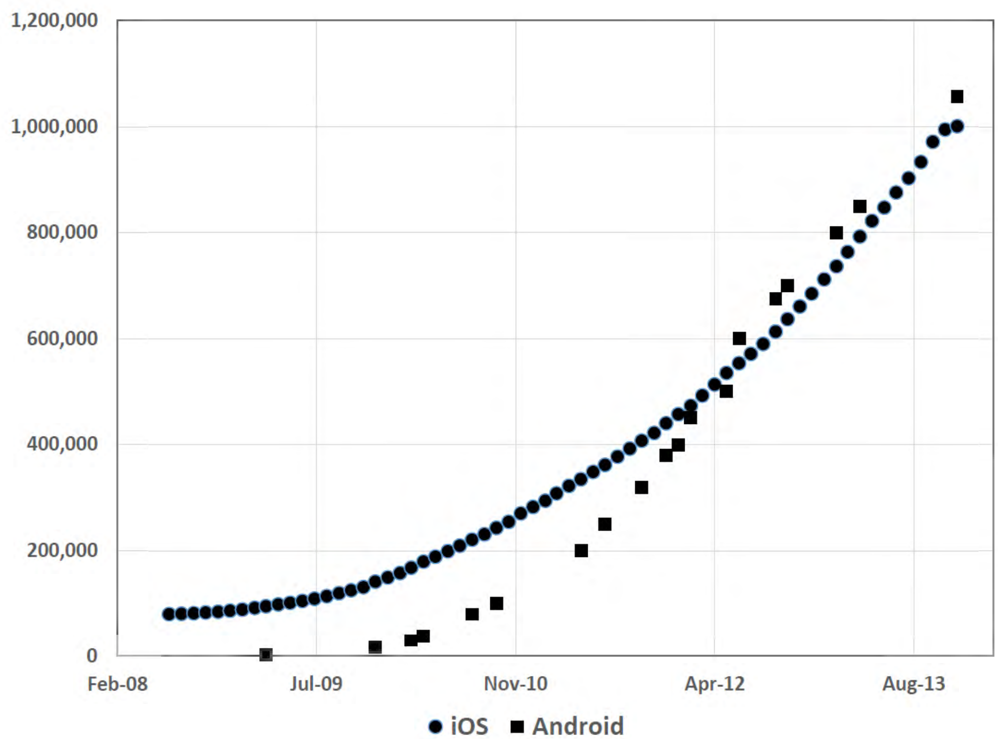


FIGURE III: TEST

continue to multihome will be robust to shocks in the relative growth rate of the platforms. This empirical explanation of the stability of the approximately split equilibrium does not appear to be in the theoretical literature on tipping.

Empirically, we identify an asymmetry between large, established firms and entrepreneurial firms, but not in the way many anticipated. Despite expectation that the massive entry due to platform sponsorship would lead to disruptive entrepreneurial innovation, we find that the most demanded apps tend to be from established firms. Our discussions with industry participants suggest that the first-order challenge for apps in the industry today is getting noticed out of the clutter of apps available. Entrepreneurial firms face a high cost of marketing to one, let alone two, platforms. Established firms are able to avoid this cost thanks to their existing customer relationships. In this setting, the potential for disruptive entrepreneurial innovation is diminished.

The paper is structured as follows: Section I. reviews the relevant literature. Section II. describes the industry setting of mobile apps. Section III. describes our model of developer choice over platforms and multihoming. Section IV. describes our data. Section V. describes how we implement the economic model to analyze our data. We discuss our findings in Section VI.. The last section concludes.

I. Literature Review

We contribute to the empirical literature on platform choice by users and complementors that relates to platform market structure. Rysman's (2004) classic study of advertising in the yellow pages identified the positive feedback between users and complementors that leads them to follow each other onto a platform. Rysman (2009) provides a valuable survey of the empirical literature. Farrell & Klemperer (2007) provide an extensive overview of the broader literature on network effects, going into the relevant theory in depth.

The largest body of empirical papers examines the economic or contractual relationships between complementors and platform sponsors, one of the core concerns of the theoretical literature. Some of these papers take up platform policies, such as openness, exclusivity, and vertical integration, in the context of platform competition (Boudreau & Hagiu, 2009; Corts & Lederman, 2009; Boudreau, 2010; Dube *et al.*, 2010; Lee, 2013). Brown & Morgan (2009) and Cantillon & Yin (2013) look at platform product differentiation in a variety of dimensions as a determinant of complementor platform choice, drawing inferences about platform market structure. Our study departs from these. In our industry, there is some vertical integration (e.g. of Google Maps with Android) but no exclusivity. There is some difference across platforms in openness, which we treat as one of the many (but not separately identified) determinants of differential app profits across platforms.

Another body of empirical papers takes up the question of whether the platform market structure will achieve the right trade-off between variety (which calls for many platforms) and size of the market / network effects (which calls for few) and the related question of whether platform competition can lead to tipping to a less than optimal platform (Church & Gandal, 1992; Dranove & Gandal, 2003; Augereau *et al.*, 2006; Corts & Lederman, 2009; Boudreau, 2012). We study similar issues, but in an industry where there is little movement over time in platform market shares so that the dynamics of supply cannot be used for identification. We are instead focused on the role of heterogeneity in the size and economic role of complementors (app developers in our case) on platform market structure.

The theoretical literature has identified a number of reasons why the industrial organization of complementor markets might impact platform market structure. Ellison & Fudenberg (2003) argue that developer's incentives to avoid competition on a platform may lessen their incentives to all supply the same platform, rendering markets less likely to tip. This topic is taken up empirically in Augereau *et al.* (2006). Bresnahan & Greenstein (1999) argue that large complementors can have influence on the choice of platforms and can direct a platform race. A number of arguments have arisen as to why and how market structure in platforms and in applications may be linked (e.g. Casadesus-Masanell & Halaburda, 2014). We make a very different argument about the size distribution of complementors. Popular complementors may have a particularly strong incentive to multihome, covering the incremental fixed costs of supplying an additional platform if they can find a sufficiently large enough number of customers to serve there. This effect will be strongest if there is little differentiation in the audiences attracted by platforms, so that a complementor popular on one platform is very likely to be popular on another. This appears to be the case for mobile platforms, where some of the most important complementors are media or entertainment products, such as games, while other of the most important complementors themselves have strong positive feedback around their own network effects. These features of the economics of the *complementors* appear to be critical to *platform market* competition. We believe we are the first to empirically document how the incentives to multihome for a few very popular apps in a fragmented market can sustain fragmentation.

We model the developers' decision to write for a platform as entry into the market. As a result, our work builds upon the market entry literature, reviewed in Berry & Reiss (2007). A large literature, descended from the work of Sutton (1991) and Berry (1992), identifies the list of potential entrants into a market as the actual entrants, present or past, in another market or markets.[1] This obviously could create a problem of selection if profit in the market at hand is correlated with profit in the market(s) used to define the list of potential entrants. We follow in this tradition, and our model explicitly addresses the problem of selection

1. Another approach avoids the problem of selection by specifying a list of niches that might be entered rather than the list of potential entrant. Examples include Bresnahan and Reiss (1991), Mazzeo (2002), and Seim (2006).

in the list of potential markets.

II. Industry Setting: Mobile Apps

An outstanding and dynamic example of the platform industrial organization is the dramatic rise in consumer use of mobile devices and the explosion of applications software running on those devices – mobile “apps." The invention of mobile app platforms has permitted developers to offer a system solution to their customers – re-using, not reinventing, the technology of mobile phones, mobile transmission, wi-fi, cloud technologies, and many other components. The newness and popularity of the mobile apps industry has created many misconceptions about the industry structure and its practices, so we use this section to clarify the facts.

The market structure of complements impacts competition between platforms. The mobile apps industry primarily serves the mass market consumer: games is the largest category (Yin *et al.*, 2014). Consumers tend to single-home, choosing only one platform as their mobile device. From a consumer perspective apps are highly skewed in their attractiveness; most, about 80%, of user demand going to about 20 of the 1.4 million apps available (Bresnahan *et al.*, 2014). The most attractive apps to users are attractive on both the main platforms, iOS and Android; relatedly, even for the next several hundred apps, there is little heterogeneity in taste for the app between the users of each of the two main platforms. We believe that these features of user demand for apps arise from the fact that mobile serves a consumer mass market.

While the platform makes it very easy to create an app, the explosion in entry by third-party developers makes it hard to gain visibility and get matched to the right customers. The platforms predominantly match apps to users through top lists. These top lists recommend apps based on aggregate lagged user downloads, employing typical collaborative filter methods to try and identify the most useful or popular apps. The most important top lists cut across app categories, since the app categorization scheme is weak. Unfortunately, since the stores and these mechanisms are effectively the only channels through which apps can reach their customers, they create incentives for apps to game the system and try to "buy" their way into the top rankings. The commercialization and distribution facilities used by the platform providers reward success for app developers with more success, and reward near misses on success with very limited visibility. Thus app developers impose positive externalities on one another through the usual platform positive-feedback mechanisms, but also impose negative externalities on one another through congestion in visibility.

Even without these distortions, simply trying to get noticed out of the close to 1 million apps competing for a users attention will require marketing expenditure for most apps, and the success of this expenditure is not assured. Our many discussions with industry participants illuminate the high costs of marketing an app.

The initial increase in the first few days after launch of the probability of being in these top lists is often fueled by apps who typically invest in marketing through "incentivized downloads" (purchasing downloads from users) and advertising their app in other apps for the express purpose of increasing their standings in the rankings. These launch campaigns average approximately $0.5 million. Despite these efforts, the probability of being on the top lists drops dramatically in the 2nd through 4th weeks, indicating that for many of the apps, these incentivized downloads and advertising campaigns did not work.

Rising above this fierce competition for visibility are a few influential exceptions. Developers with existing lines of business and customer relationships outside of mobile (e.g., Facebook, Starbucks, United Airlines) can get around the top list bottleneck, thanks to their existing customer relationships. These apps come primarily from third-party developers (Google Maps being an exception) and are monopolists, in the sense that everyone uses them. It would therefore be extremely expensive for a platform provider to pay the developers to make them exclusive.[2] The market structure of these complements imply multihoming by the most attractive apps and therefore a strong tendency for app supply to contribute to the stability of a platform equilibrium approximately split between iOS and Android.

The same market structure facts are entirely consistent with the tip away from the Blackberry and from the Windows mobile platform. The Blackberry platform was dominant in smartphones in an earlier, business-user oriented era (see Bresnahan & Greenstein (2014) for more complete analysis of this incident and for cites to industry sources). Blackberry had almost all of the users and apps of the earlier era, but the size of that installed base was overwhelmed by the mass-market consumers served by the Apple iPhone's and its entertainment and media apps. Blackberry switched to a more consumer friendly strategy after a period of time, but found itself in a downward tip with too few users to attract apps (even the most popular multihomers on iOS and Android) and too few consumer-oriented apps to attract users. Microsoft, a perennial late entrant into mobile platforms, found itself in similar circumstances from just before the launch of WIndows Phone 7 until recently. To deal with the shortage of apps, Microsoft has been paying small bounties to each developer who submits an app, and very large bounties, reportedly up to $100k, to selected developers (Bass & Satariano 2010). Windows tablets, which share apps to some degree with Windows PC, have succeeded considerably better without the bounties. This very expensive program has been sufficient to keep Windows phone from completely dying off, though its shipments are about 1/20th of those of Android. At this stage, the market has strongly tipped away from both Blackberry and Windows Phone.

A review of the market structure of the complements in the mobile app industry clearly reveal that

2. Indeed, the platforms have thus far only succeeded in convincing some anticipated popular games to be exclusive for a short period of time at initial launch but not long-term.

developers and users can both tip away from platforms in this market and sustain fragmented market shares as an equilibrium. In the model and estimates that follow, we will examine the multihoming mechanism for sustaining a fragmented equilibrium.

III. Economic Platform Model

Like all models of platform market equilibrium, we start from a model of developer and user platform choice. The developer and users models differ both economically and econometrically. In this section, we lay out the economic models. We are particularly interested in platform market is "tippiness," in the language of Farrell & Klemperer (2007), and in the conditions for stable equilibria that are balanced across platforms.

III.A. Developers

Developers can multihome. They choose on which of platform(s) p to publish their app. Only by publishing an app on a platform can a developer reach the users who choose that platform: A developer who publishes only an iOS app can sell to iPhone users but not to Android, Blackberry or Windows Phone users. We model developer choice as deciding which applications market(s) to enter, where the market boundaries are defined by the platforms. Like all entry models, our model assumes that the developer has fixed costs and will only produce if the size of the market is sufficiently large to cover them.

We model variable profit for each app as a linear function of the number of customers, following entry models generally, and label the variable profits per customer on platform p as M_p. For platform economics, it is important to distinguish between U_p, the number of users of platform p, and r_p, the app's "reach" on platform p, measured as a the fraction of U_p that use the app. The number of customers for the app is $U_p \times r_p$, but the two parts have different economics. The overall attractiveness of the platform determines U_p, while the attractiveness of the particular app itself determines r_p. The variable profit of an app on a platform is $\pi_p = U_p \times M_p \times r_p$.

Finally, let the fixed costs of publishing an app on one or more platforms $\{p\}$ be $C_{\{p\}}$, with (for example) $C_{\{d,i\}} = C_d + C_i$. This is an economic assumption that the platforms are very different, i.e. that there is no fixed-cost savings from writing for both. In our context, we make this assumption because of what developers have told us about their cost function: porting is not extremely difficult technically. Instead, platform-specific marketing costs, such as buying downloads from non-overlapping populations of users, are large and distinct to platforms. We normalize the costs and revenues of supplying for no platform to 0.

We now introduce a notation for a particular app, a. Platform supply for a developer is simply the

choice of the set of platforms to supply:

$$\max_{\{p\}} \sum_{\{p\}} U_p \times M_p \times r_p - C_{\{p\}} \tag{1}$$

If an app is supplied to a platform, we denote this as $S_p = 1$; $S_p = 0$ otherwise. We note two features of this supply behavior relevant to the platform market. Developer supply to a platform is increasing in the number of users of that platform U_p – this is the typical structure of platform markets. Second, a developer with a particularly attractive app (large r_p) has more incentive to supply a platform, and a developer with an app that is particularly attractive to the users of multiple platforms (large r_p and large $r_{p'}$) has more incentive to multihome.

The supply behavior in Equation (1) determines the set of apps $A_p()$ available for each platform. If we let U be the number of users of all platforms, $A_p(U)$ is determined by Equation (1). In general, we expect $A_p()$ to be increasing in U_p.

In Section V. below, we will add structure to the problem in Equation (1) and estimate it econometrically, thereby coming up with quantitative estimates of the profit function for developers and of $A_p(U)$.

III.B. Users

Users choose a single platform. User j picks the platform which has the highest utility to her, given by $V_{jp} = v_{jp} + u(A_p)$ where v_{jp} is the intrinsic value of the platform to user j (based on platform services, ease of use, the price of the platform good, etc.) and $u(A_p)$ is an index of the value of the apps available to users on platform p. We assume that $u(A_p)$ is weakly increasing if apps are added to the set A_p. Then the number of users who choose each platform can be written as a function of the apps available on each platform. This function $U(A) = U_p(A), U_{p'}(A)$ is determined by $U_p = \|j|v_{jp} + u(A_p) \ \geq v_{jp'} + u(A_{p'}) \ \ \forall p'\|$ for all p.

Under these assumptions, U_p will not decrease if an app is added to A_p (and will not increase if an app is added to $A_{p'}$ $p' \neq p$.)

III.C. Index numbers of user value of apps on a platform

Our model of users will not lead to a user choice model for estimation but instead to an index number of the value of the all the apps available to consumers on a platform $u(A_p)$. Because of the approximate tie in apps across the two largest platforms, Android and iOS, and the very limited attractiveness of the two smaller platforms, Windows Mobile and Blackberry to consumers, there is little variation in the data to estimate an econometric model of user choice.

Since most apps are free, a price-index approach to valuation is not feasible. We instead employ a quantity index of app valuation per customer.[3] We calculate quantity based on "reach," r_{pa} , the fraction of users on platform p who choose to use app a. Thus we define the quantity index $u^Q(A_p) = \sum_{a \in A_p} r_{pa}$. The idea of this index is that those apps which attract many users make a large contribution to the attractiveness of the platform.

Like any assumption of the functional form of demand, an index number formula embodies unverified assumptions. We cannot estimate user demand, especially not the part relating demand to the availability, variety, etc., of apps. Thus we simply make different quantitative assumptions about the index number formula. Accurately approximating the index number is rendered somewhat easier in our industry, however, by the extremely high concentration of app market share. This is a mass market, and many apps themselves have network effects (e.g. Facebook.) Accordingly, the measurement task faced by our index number formula is to capture the relatively small number of highly attractive apps.

Users choose a single platform in our industry, in contrast to developers who can and do multihome. From an indirect networks effects perspective, what matters is the aggregate demand for a platform by users, so we focus our user modeling on an index of the contribution of the apps available on a platform to the attractiveness of a platform to a large number of users. Thus we now shift our focus from each app to all apps. The set of apps on a platform, N_p, and in an obvious notation we say that app a is available on a platform if $a \in N_p$ and that app a has reach r_{pa} on p. Since most apps are free, the usual index number problems for discrete choice are not present (Small & Rosen, 1981). Reach is a measure of the quantity demanded of an app, in the specific sense of the fraction of users who choose to use the app. Accordingly, we measure the contribution of all apps to the total attractiveness of a platform as $\sum_{a \in N_p} r_{pa}$.

III.D. Platform Market Equilibrium

A platform market equilibrium is a U and an A such that $U = U(A)$ and $A = A(U)$. Platform markets will tend to tip in the sense of Farrell & Klemperer (2007): positive feedback tends to lead to a dominant platform(s) rather than to more or less equal market shares. The conditions for a market to tip are linked to the conditions for any equilibrium with more or less equal market shares *not* to be "stable." Here, stability refers to the "dynamic" defined by $U = U(A(U))$.[4]

The stability conditions approach has produced a number of useful results. The platform market will tend to fragment rather than tip if (1) indirect network effects are just not that important, i.e. (1a) users

3. See Small & Rosen (1981) who deal with the appropriate quantity index when prices are varying.

4. The equilibrium is stable if all the eigenvalues of the Jacobian of the function $U(A(U))$ are less than one in absolute value. Consider the simple case in which there are only two platforms and every user uses one of them, so that we can write $x = U_i$ and calculate U_d as $M - x$, where M is the total number of users. This is the case most freqently seen in the theoretical literature and is thus the most familiar. In this simple case, an equilibrium is an x such that

do not care much about the presence of apps and (1b) app developers do not care much about users. If indirect network effects do matter, the stability conditions approach takes a more quantitative turn. The market will tend to fragment rather than tip if (2) users have strong and varied platform tastes, i.e. if the variance across j of $v_{jp} - v_{jp'}$ is large enough to overcome the network effects, and/or if (3) developers' platform profits vary strongly across developers, i.e. if the variance across a of $\pi_{ap}- \pi_{ap'}$ is large enough to overcome the network effects.

Our results add a new element to this list, i.e. that a highly skewed distribution of developer attractiveness to consumers can lead to a situation in which influential developers choose to supply more than one platform and would maintain that choice even if the relative user market shares were to shift somewhat. This is in many ways the opposite of condition (3). Rather than having strong platform preferences that overcome the pull of the network effects, developers inframarginally choose to multihome, rendering the trade-off between access to apps and native platform preferences irrelevant for users. The market is stable at a fragmented platform market structure because (a) the most attractive developers multihome and would continue to do so if there were a modest deviation in user shares by platform and (b) therefore users can choose platforms based on their $v_{jp} - v_{jp'}$ without need to trade off against the $u(A_p) - u(A_{p'})$.

Some of the existing explanations can be discarded based on our estimates below in Section VI., and others can be discarded on the basis of industry facts presented earlier in Section II.. Using our estimates, we will discard the possibility that (1b) app developers do not care much about users and (3) developers' platform profits vary strongly across developers. An examination of the history of mobile platform markets dispels the hypothesis that (1a) users do not care much about the presence of apps. Users care enough about the availability of apps, and app developers care enough about the number of users, that tipping is a real possibility, as evidenced by the tip away from Blackberry and the expensive investments being made by Windows to pay for multihoming by developers to its platform despite its small market share. The tip away from Blackberry due to small market share and app availability and grand efforts by Microsoft to stay alive by subsidizing apps leads us to discard the potential explanations for fragmentation that users care little about apps and/or developers.

$$\begin{aligned} A_d &= A_d(M-x, x) \\ A_i &= A_i(x, M-x) \\ x &= \| j | v_{ji} + u(A_i) \geq v_{jd} + u(A_d) \| = \chi(A) \end{aligned}$$

The condition for stability is that at $x = \chi(A(x))$, the function $\chi(A(x))$ cuts the 45% line ($x = x$) from above.

IV. Data

We have gathered a sample of apps which were written for either iOS (iPhone), or Android phones, or both. We gathered our primary data set from public sources with the help of research assistants. We asked them to download each app in the sample and to fill in a questionnaire with 200 questions regarding the app's use of advertising, in-app purchases, and other monetization strategies. We also asked them to visit the developer's website to learn about the app's platform availability and developer characteristics.

We match these data to the January 2013 Mobile Metrix dataset from comScore, filling out our questionnaire for apps as necessary to ensure full coverage of the comScore apps. Two panels, one with Android phones and the other with iPhones, of approximately 5,000 US adult users, allow comScore to track their possession and usage of apps. comScore reports their data aggregated to the app*platform*month level. In this data, however, we only observe apps which meet a minimum usage test for each month on each platform: comScore includes data on the app only if it is used on that platform by more than 5 (at least 6) unique users.[5] As a result, while we may know from our primary data set that $S_p = 1$, if the app does not meet the usage criterion on a particular platform, then nothing about it is reported on that platform, i.e., $S_p^* = 0$, where S_p^* denotes the supply of an app on a platform as reported by comScore. We further drop apps produced by Apple, Google, carriers, and OEMs, which are typically pre-loaded onto devices, so that an observation is an app developed by an independent software vendor. This results in $1,044$ apps.

We observe S_p, S_p^*, and r_p^*, comScore's report of its estimate of the app's "reach" for each platform on which the app has met the usage criterion.[6] Note that r_p^* is censored by the comScore sampling rule that it is not reported unless $S_p^* = 1$. This set of indicator variables concerning the developer's publication decisions and the app's satisfaction of comScore's usage criterion, together with the reach of the app when that criterion is met, form the dependent variable in our model.

Figure IV shows how the total value to users of apps on a platform, $\sum_{a \in N_p} r_{pa}$, is distributed among the top N apps (as ranked by r_p^*). The first set of bars on the left indicate that the top 10 apps on iOS and Android already account for 30% and 25%, respectively, of the total value of all apps on their platforms. The second set of bars show that a five-fold increase in the top app cutoff only doubles the coverage of the value of all apps. There is diminishing marginal returns to the extra value provided by the marginal apps. This market is very skewed on both platforms, with most apps reaching few users and a very few apps reaching almost all users. Very few apps are providing most of the value for either platform.

Figure V shows the distribution of supply of apps and observed supply of apps on comScore, where S

5. comScore actually tracks apps with less than this level of usage if a client has requested tracking. We drop these apps to ensure uniform truncation of observations from the comScore dataset.

6. comScore projects their data onto US census data to correct for selection in their panel before reporting r^*.

Figure IV: Proportion of Platform Value from Apps to Users from the Top N Apps by Platform. Source: comScore, January 2013.

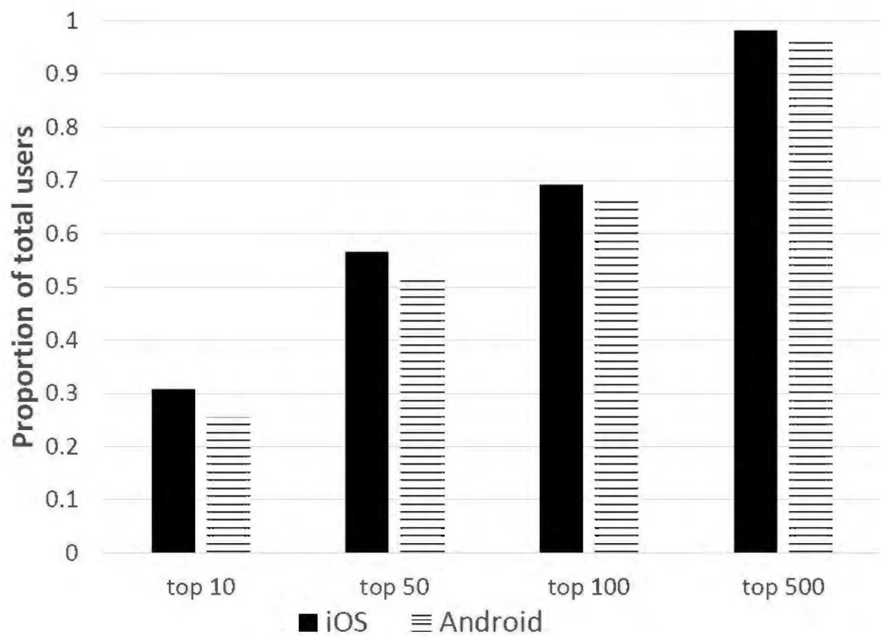


Figure V: Platform Choices (S, S^*) with b= both, d=Android, i=iOS.

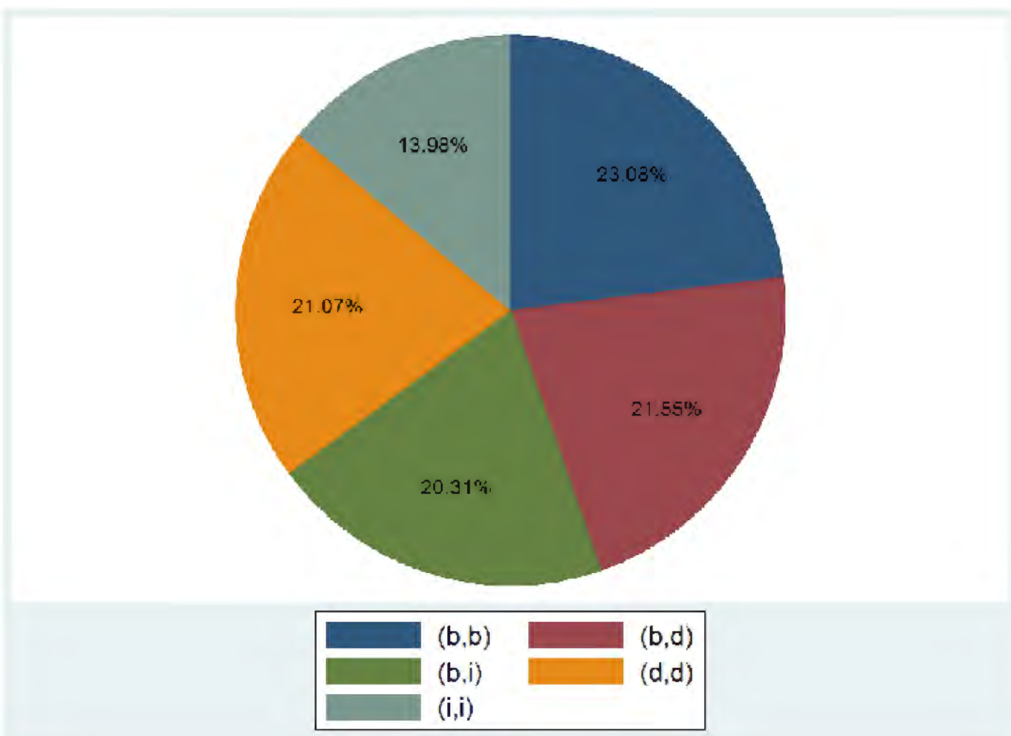


TABLE I: DESCRIPTIVE STATISTICS OF S, S^*, AND r^*; 1,044 APPS

(S, S^*)	mean r^*	st. dev. r^*	min r^*	max r^*	N
$(b,b) : r_i^*$	.035	.081	.001	.775	241
$(b,b) : r_d^*$	.030	.069	.001	.735	241
(b,i)	.010	.012	.001	.122	212
(b,d)	.010	.014	.001	.086	225
(i,i)	.014	.056	.001	.664	146
(d,d)	.014	.043	.001	.605	220
overall: r_i^*	.021	.060	.001	.775	599
overall: r_d^*	.018	.049	.001	.735	686

and S^* take on the values of b, d, and i to indicate whether the app was multihomed or single-homed for Android or iOS, respectively. Each slice of the pie chart shows one value of (S, S^*). The first letter in each pair indicates whether the app was written for both platforms or only Android or the iPhone, based on data collected by our research assistants. The second letter in each pair indicates whether the app had enough quantity demanded to be observed in comScore on both or only one of the platforms. For example, the notation (b, d) means that the app was written for both platforms but crosses the comScore threshold only on the Android platform. A substantial 36% of apps do not multihome. Despite the fact that 64% of apps do multihome, only one-third of those actually reach enough users on both platforms for us to observe r_d^* and r_i^* in comScore. Descriptive statistics for the observed reach for each slice of the pie are given in Table I.

The characteristics associated with app a (or its developer), X_a, comprise the regressors in our estimation. All our regressors are indicator variables; descriptive statistics are provided in Table ??. We use information about whether the developer is publicly traded and the developers' other lines of business to indicate whether the developer is an established firm or an entrepreneur. Developers can be in one of three mutually-exclusive categories: mobile-only business (e.g., Rovio Entertainment, maker of Angry Birds), online business (e.g., Facebook), or offline business (e.g., Delta, Nike, other brick and mortar stores selling physical goods or services). Publicly traded firms that have another line of business either offline or online are considered established firms. Firms that are not publicly traded and are mobile only are considered entrepreneurs. Although 54% of apps exhibit advertising and 29% of apps use in-app purchases (IAP), it turns out that publicly traded apps are almost perfectly correlated with apps with no advertising and no IAP, so we only use publicly traded in our final estimation.[7] We abbreviate the 12 types of apps/developers produced from the combination of these regressors as designated in the first column of Table III, and note the number of each type that appear in our sample in the last column.

7. We did use both ads and IAP with no significant contribution in our results.

TABLE II: DESCRIPTIVE STATISTICS OF X; 1,044 APPS

X	Percent of Apps
game	27%
publicly traded	37%
mobile	42%
online	29%
offline	29%

TABLE III: NUMBERS AND TYPES OF APP-DEVELOPERS IN SAMPLE

abbreviation	Lines of Business	Category	Ownership	observations
MGP	Mobile Only	Game	Private	177
NGP	Online	Game	Private	48
FGP	Offline	Game	Private	22
MGT	Mobile Only	Game	Publicly Traded	3
NGT	Online	Game	Publicly Traded	24
FGT	Offline	Game	Publicly Traded	53
MOP	Mobile Only	Other	Private	257
NOP	Online	Other	Private	151
FOP	Offline	Other	Private	76
MOT	Mobile Only	Other	Publicly Traded	1
NOT	Online	Other	Publicly Traded	80
FOT	Offline	Other	Publicly Traded	152

V. Econometrics of Demand and Developer Supply

We estimate the parameters of user demand for apps on each platform, developer signals about demand, and the decision rule determining supply of an app to each platform. Our dependent variables are S, S^*, and r^*.

Our econometric model of developer supply also deals with the problem of selection of potential entrants by accommodating both the parts of the sampling frame about which we have information, e.g. comScore's "panel," and the parts about which we are not informed, such as heterogeneity across developers.

In this section we describe

- Our sample selection rule (since we collect our own data, we have some control over this).
- Our choice of functional form.
- Our approach to the profitability signal received by the developer.

These modeling decisions are made jointly to permit closed-form calculation of the likelihood in unselected and in selected samples. These modeling decisions are also made to make sure our most important inferences are driven by the data rather than modeling artifacts; we leave much of that identification discussion to the results section (Section VI.).

V.A. Distribution of Developer Signals and Realized App Success

We model demand for the same app as potentially dependent over platforms, but also allow user preferences to be different. This lets us test the hypothesis that users who have chosen different devices (iPhone or Android Phone) have the same app demand.[8] In what follows, we work backward from comScore, which is the source of two of our three dependent variables.

V.B. Observables conditional on developer decisions and app demand

Since we know the comScore sampling frame, it is easy to work out the sampling distribution of S^* and of r^* conditional on S and r, the supply decision of the developer and the realized demand behavior of the users. As a threshold point, the comScore separately samples iOS and Android users, so we work out the independent distribution on each platform.

8. We have no model of device choice, so cannot test hypotheses about user preferences for devices. A large commercial literature establishes that very different users chose iPhone, most notably richer ones (see discussions in Bresnahan *et al.*, 2014).

comScore has a sample of 5,000 users. Let g be the number of platform p users that have the app in the comScore sample. The density of g conditional on r_p is *binomial*, $f(g|r_p) = \binom{5000}{g}(r_p)^g(1-r_p)^{5000-g}$ so the probability of the event $g > 5$, can be easily calculated. Call this probability (which is one minus the *binomial* CDF evaluated at $g = 6$) $F_g(r_p)$. $F_g(r_p)$ is $\Pr(S_p^* = 1|r_p)$. Given this binomial structure, we naturally model the distribution of r_p across apps as *beta*. This makes the sampling distribution of S_p^* beta-binomial. It also gives us the sample distribution of observed reach on each platform $(g/5000)$.

We assume that developers' signals about the demand for their app are equal to potential reach $\widetilde{r} = (\widetilde{r}_i,\ \widetilde{r}_d)$. This does not mean the developer's signal is highly informative, since we are introducing a gap between potential reach and realized reach.[9] We make functional form assumptions about $\widetilde{r}_i$ and $\widetilde{r}_d$ that give them marginal *beta* distributions but that need not be independent. To do this, we assume a mixture model built up from underlying *beta* distributions. There are three independent *beta* random variables, which we call q_i, q_d and q_b. Realized reach, and therefore the developer's signal, is either equal to the independent pair (q_i, q_d) or to the perfectly dependent (q_b, q_b).[10]

$$(\widetilde{r}_i,\ \widetilde{r}_d) = \begin{matrix} (q_b, q_b) & \text{with probability } \omega \\ (q_i, q_d) & \text{with probability } 1-\omega \end{matrix}$$

Further, we assume that $q_i \sim \text{beta}(\alpha_i, \beta_i)$ and $q_d \sim \text{beta}(\alpha_d, \beta_d)$, and that q_b is distributed beta with parameters that give it the mean and variance which are the average of those of q_i and q_d.[11] So the parameters to be estimated here are α_i, β_i, which we interpret as the iOS users' app demand parameters, and α_d, β_d, which we similarly interpret as the Android users' demand parameters, and ω.

Our second modeling choice is the relationship between the developer's pre-entry signal and post-entry success. We reject the simplest model in which these are perfectly correlated based on industry facts described above, such as the frequency of apps introduced with a large marketing push in the app stores, that are not ultimately successes. Our specific model of less-than-perfect dependence between signal and ultimate success draws the distinction between the reach the app would have if it were visible to all customers by being on the top list in the app store, potential reach, $\widetilde{r}_p$, and the reach it ultimately achieves, realized reach r_p.

Not all apps can appear on the app store's top list, and we model the probability that an app appears

9. We have three concepts of reach on each platform: signal, realized, and potential. We observe a (censored) variable close to realized reach, and the difference between S^* and S tells us something about the distinction between signals and actual reach. That is, the data permit measuring at most two concepts of reach. We adopt the normalization that potential reach is less than perfectly correlated with realized reach, but potential reach is identical to the signal received by the developer. In another paper, we use the daily success of newly introduced apps in an effort to learn more about the developers signals.

10. There are several other ways to model correlated *beta* distributions. The Sarmonov method is inappropriate for our purposes, since it limits the correlation to be near zero. Another method is to build up the distributions from ratios of *gamma* distributions. This, however, does not lead to both marginal and conditional *beta* distributions and thus would leave a number of the calculations below much more difficult.

11. We assume that $\mu_b = \frac{\alpha_b}{\alpha_b+\beta_b} = \frac{\mu_i}{2} + \frac{\mu_d}{2}$ and $\sigma_b^2 = \frac{\sigma_i^2}{2} + \frac{\sigma_d^2}{2}$ in our estimation.

on the top list as $A(\widetilde{r}_p, \tau)$. Here $\widetilde{r}_p$ is the fraction of platform users that would choose the app if they knew of it and τ is a parameter that indexes the efficiency of the app store at matching. Similarly, we introduce another random variable, $\overline{r}_p$, the reach the app would gain if it were relegated to the depths of the app store rather than being on the top list. We assume that this is a "shrunk" version of $\widetilde{r}_p$, i.e. that $\overline{r}_p$˜$beta$ and that $E[\ \overline{r}_p] = \delta E[\ \widetilde{r}_p]$ and $var[\ \overline{r}_p] = \delta var[\ \widetilde{r}_p]$, with δ a new parameter to be estimated.

These assumptions mean that the effective demand for the app, r_p, is given by

$$r_p = \begin{cases} \widetilde{r}_p \text{ with probability } A(\widetilde{r}_p, \tau) \\ \overline{r}_p \text{ otherwise} \end{cases}$$

where

$$A(\widetilde{r}_p, \tau) = \widetilde{r}_p \mathbb{I}(\widetilde{r}_p < \tau) + \mathbb{I}(\widetilde{r}_p \geq \tau), \tag{2}$$

and $\tau \in (0, 1)$.[12]

The variable r_p determines the developer's potential profits on platform p, and if the developer does enter, r_p is the random variable censored to determine S_p^* and, r_p^*. Therefore, the app is written for platform p if and only if the following condition is satisfied:

$$\mathbb{E}[r_p|\widetilde{r}_i, \widetilde{r}_d] = \mathbb{E}[\widetilde{r}_p A(\widetilde{r}_p, \tau) + (1 - A(\widetilde{r}_p, \tau))\overline{r}_p|\widetilde{r}_i, \widetilde{r}_d] \geq \kappa_p, \tag{3}$$

where $\kappa_p = C_p/M_p/U_p$. This is simply a normalization. We will estimate κ; the normalization determines the interpretation. In our application, U_i is slightly smaller than U_d, so if we find that $\kappa_d > \kappa_i$ the interpretation is either that the fixed costs of writing an Android app are higher $C_d > C_i$ or that the per-customer profits of an Android app are lower $M_d < M_i$. We cannot distinguish these two hypotheses.

This gives us the distribution of the bivariate supply dummy S, and we saw above how we get the distribution of S^* and r^*.

V.C. Parameters and Likelihood

The parameters of the model, which together we call θ, are $\alpha_i, \beta_i, \alpha_d, \beta_d$, ω, τ, δ_i, δ_d, κ_i, and κ_d. All of these parameters will, in some specifications, vary with the regressors X_a. Above, we showed the conditions for observing r_p^*, and the sampling distribution of r_p^* conditional on r_p. Similarly, we showed the probability that S_p^*,= 1 conditional on r_p. Finally, we showed the event determining S, which is a crossing condition for

12. We attempted alternative functional forms for $A()$. This form works remarkably well empirically, and captures the point, made to us by many industry participants, that the app stores are harder on the less successful than on the successful.

beta random variables. Under our assumptions, the distribution of r_p is beta, so we can calculate (using the beta binomial distribution) the likelihood of r^* and S^* and the probability of S. The joint distribution of r^*, S^* and S is the likelihood $L(S,\ S^*,\ r^*|X_a,\ \theta)$. We can calculate it in closed form given our assumptions, of which the most important for this purpose are the beta functional form and the structure of the dependence across platforms.

It is not hard to see how our model is identified and the role of the functional form of the distribution in it. Within a cell defined by X_a the distribution of $(r_i,\ r_d)$ has *beta* marginals that depend on $\alpha_i, \beta_i, \alpha_d, \beta_d$, ω, τ, and δ. The distribution of $(r_i,\ r_d)$ are easily recovered by predicting the sample distribution of $(r_i^*,\ r_d^*)$ conditional on observing them. To get from the distribution of $(r_i,\ r_d)$ back to the underlying distribution of $(\widetilde{r}_i,\ \widetilde{r}_d)$ and the mixing parameters, we use the probability of the events $(S,\ S^*)$ shown in Figure V.

The structure of the dependence among $(\widetilde{r}_i,\ \widetilde{r}_d)$ appears limiting, because it is a mixture model, but for predicting S and r^*, this is not an issue. The event $S_{p'} = 1$ occurs not only when $(\widetilde{r}_i,\ \widetilde{r}_d) = (q_b,\ q_b)$, but also when both of the latent variables $\widetilde{r}_i$ or $\widetilde{r}_d$ takes a large value.

V.D. Sample

Since we collect our own data from primary sources, we have some control over the definition of our sample. It is not possible to design a sample which is entirely free of selection bias: app developers who have had an idea but who have failed to publish their app, for example, cannot in principle be observed. We can, however, minimize the degree to which sample selection bias correction leads us to introduce new assumptions.

Our sample consists of apps with $S^* \neq (0, 0)$, that is apps which appear in comScore on at least one platform. The rationale for this is simple, if novel. Given that we only observe the censored variable r_p^* if $S_p^* = 1$, we must introduce a model of the event $S_p^* = 1$ in order to deal with the censoring. Under our sampling scheme, observed potential entrants into platform p' (the "other" platform) are those who appear in the comScore data on platform p ("this" platform.) With an explicit model of the app appearing on "this" platform, we can correct for the selection bias in our treatment of entry into the "other" platform. Of course, we treat both platforms symmetrically and examine the joint distribution of S, S^*, and r^*.

The two sets of two dummy variables S and S^* define 9 possible states (combinations of whether an app is available on each platform and whether it has achieved the demand threshold to be observed in comScore for each platform). Because of our sampling scheme, however, we only observe 5 of these states. These are the 5 cells in Table IV which contain at least one r_p^*. The reach for each state is presented in each cell of the table. Since this sample definition is conditional on the dependent variable of app supply to platforms,

TABLE IV: APP SUPPLY STATES

d, i	$S_i = 0$	$S_i = 1$	$S_i^* = 1$
$S_d = 0$		r_i	r_i^*
$S_d = 1$	r_d	r_d, r_i	r_i^*
S_d^*	r_d^*	r_d^*, r_i	r_d^*, r_i^*

we must also redefine the likelihood to adjust for this conditioning, a topic we address in a moment.

Entry models generally need to condition on something in order to define the set of potential entrants. The problem is that firms that have not entered any market do not exist and that data on them cannot be gathered. A variety of solutions have grown up to deal with this. Berry (1992), and many papers following in that tradition, define the set of potential entrants into a particular market by observing firms that have already entered related markets. In his example, potential entrants into a particular airline city pair market are firms serving other city pair markets already. This sample definition conditions on the lagged dependent variable in an adjacent market. One could hope that the double attenuation that comes both from lagging and from looking at an adjacent market renders any sample selection bias small (the most common solution) or one could attempt to correct for sample selection bias. Another solution is to avoid defining a set of potential entrants, but instead examine a fixed set of market niches each of which could (or could not) be filled by a particular entrant. Bresnahan & Reiss (1991) use this approach, as does Mazzeo (2002) who looks at entry in the categories of high-quality motels and low-quality motels. The Bresnahan & Reiss approach only works when one can define the market niches a priori, so we cannot use it. The Berry approach, with a correction for the sampling process, works for us because we observe a parallel market, i.e., for entry into iOS we look at hit apps on Android.

Our model can be understood as having many of the features of a "Type II" Tobit. This familiar model is written $y_2 = \begin{Bmatrix} y_2^* \text{ if } y_1^* > 0 \\ 0 \text{ if } y_1^* \leq 0 \end{Bmatrix}$, where y_2 is a variable, like reach on one platform for app a for us, that is only observed if an event, labeled in the Tobit as $y_1^* > 0$, occurs.

A simplified version of our model would have exactly this structure. Suppose we were only studying entry into platform d, and we gather our sample of firms that have entered and been successful on platform i. Then we would let $y_2^* \equiv r_d$ and introduce $y_1^* \leq 0$ as a model of sample selection. If these two random variables (demand on platform d and entry and success on platform i) are correlated, as one would expect, then we would use the Type II Tobit to deal with the relevant sample selection problem.

This simplified version is how we solve a classic econometric problem in models of entry. If the set of potential entrants into one market are actual entrants in another market, and if profits in the two markets

are correlated, there is a sample selection problem.

We use this same sample selection logic but embed it in the structure of our model, which involves a few changes in the specifics. First, our model is symmetric across the two platforms, so that there are two y_2^* variables, and we have different conditions for observing the two y_2 variables. We have two variables like y_2: r_d and r_i are both observed only conditional on events. Second, in our model, $y_1^* > 0$ (the condition for observing r_d or r_i) is a compound event, depending not only on the realization of reach but also on the app being included in our sample. We observe r_d if $r_d^* > .001$. Third, we observe not only the truncated reach, but also the fact of writing for the platform or not; these events are observable whenever the app is part of our sample. None of these complications changes the nature of our approach to sample selection profoundly.

We choose a symmetric (across platforms) sample selection rule, and thus study entry into (and success on) the markets defined by both platforms simultaneously. We have both economic and econometric reasons for taking this approach. From an economic perspective, our platform focus leads us to studying multihoming, i.e., entry into both platforms, which requires the symmetric treatment. From an econometric perspective, modeling both platforms at once is our best identification strategy. We have no instruments for the "selection equation," so we need to put economic restrictions on the conditioning event. In our approach, each conditioning event has a full economic model, since it is part of our dependent variable.

V.E. *Sampling Correction*

We start from the universe of potential app developers who have an idea for an app. We have already defined the unconditional likelihood, $L(S,\ S^*,\ r^*|X_a,\ \theta)$. We have already shown how to calculate the probability of the events $S_p^* = 1$. Using the same logic, we calculate the probability, $F_S(S,\ S^*,\ r^*|X_a,\ \theta)$ that either $S_d^* = 1$ or $S_i^* = 1$ or both. Then we maximize the conditional likelihood

(4) $$L_C(S, S^*, r^*|X_a, \theta) = L(S, S^*, r^*|X_a, \theta)/F_S(S, S^*, r^*|X_a, \theta)$$

VI. Results

We focus our results discussion on the results related to the platform economics hypotheses. First, we examine differences between iOS users' app preferences and Android users' app preferences. We then examine the platform choice by developers to explain the stability of a fragmented market. Interestingly, when a platform is already split, similar user app preferences create a strong incentive to multihome for the

most demanded apps, sustaining the market split.

VI.A. App Demand on iOS vs. on Android

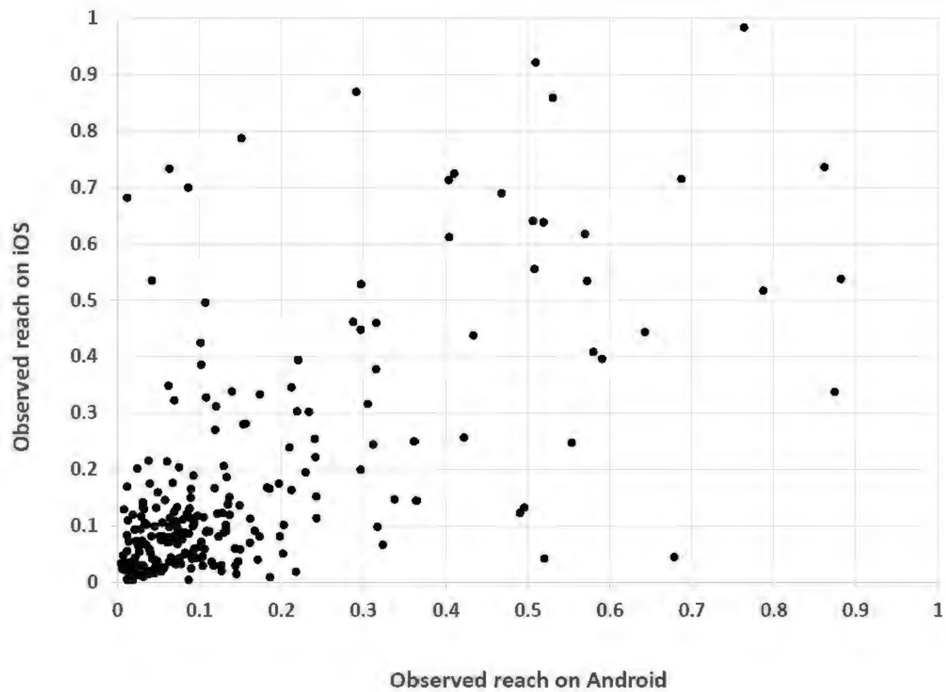
Our model permits app demand to vary between iOS and Android users by specifying a number of platform-specific parameters: $\alpha_i, \beta_i, \alpha_d, \beta_d$. Recall that we assumed about demand that the distribution of $q_p \sim \text{beta}(\alpha_p, \beta_p)$ for $p = a, d$ and that both α and β are, for each p, regressions on X. Since X takes on only 12 values (the different developer-app-types see Table III) we define α_{pX} and β_{pX} as the values for platform p in all the cells of X. We report estimates of these in Tables VII through IX. Bootstrapped 95% confidence intervals are reported in parentheses beneath each estimate. There are no restrictions in our model on these parameters between the two platform, so we first test the hypotheses that the parameter vectors are the same across platforms: $(\alpha_{iX}, \beta_{iX}) = (\alpha_{dX}, \beta_{dX})$. Bootstrapped tests of statistical significance between the estimates across platforms indicates that we cannot reject equality.

This follows partly from the fact that parameter values are similar and partly from the fact that some parameters are imprecisely estimated, but mostly the former. When we examine the means of q_i, q_d at all the different values of $(\alpha_{iX}, \beta_{iX})$ and $(\alpha_{dX}, \beta_{dX})$, the mean percentage absolute difference differ across platforms is only 3%. Predicted demand, in the sense of demand predicted by X, is very similar across platforms. In contrast, predicted demand varies strongly across X. If we instead test the hypothesis that $(\alpha_{pX}, \beta_{pX}) = (\alpha_{pX'}, \beta_{dX'})$ for all p, X, X', we easily reject. Mean demand is the same across platforms for the same kind of developer/app, but varies within platform for different kinds of X.

The tendency of a given app to have very similar potential demand on one platform to what it has on the other is stronger than the other is somewhat understated by the coincidence of $(\alpha_{iX}, \beta_{iX})$ and $(\alpha_{dX}, \beta_{dX})$. Our model also permits dependence across platforms in user's demand for a particular app conditional on X through the parameters ω. Our specification permits ω to vary with X; estimates are in Tables VII through IX. All estimates of ω are above 0.8 for Online and Offline firms that are Publicly Traded. Mobile Only firms consistently have the lowest estimated ω. The mean of ω overall is .55; that is, conditional on X, we estimate that there is a great deal of dependence in $(\widetilde{r}_i, \ \widetilde{r}_d)$. The correlation between r_d^* and r_i^* when both are observed is 0.60. When we restrict attention to those apps for which the demand threshold is raised to 1%, the correlation is 0.86. This correlation identifies ω.

The parameters τ, δ_i and δ_d allow for frictions in the market that may cause expected realized demand to be less than potential demand. The parameter τ enters the probability that an app can cut through the clutter of an app store and be seen by all its potential customers. The parameters δ_i and δ_d, account for the losses associated with not cutting through the app store and define how much lower is expected realized

FIGURE VI: JOINT DISTRIBUTION OF OBSERVED REACH r^* FROM COMSCORE JANUARY 2013 ON ANDROID AND IOS FOR $S^* = b$.



reach than potential reach.

The τ threshold is estimated to be 0.87; the maximum reach observed on either platform in our sample is only 0.775. This means that for most apps in our model with very high probability $r < \widetilde{r}_p$. The expected realized reach on both platforms is only 20% ($\delta_i = .21$, and $\delta_d = .19$) of expected potential reach. The high value of τ and low value of δ mean that observed (realized) reach is significantly lower and less correlated across platforms than is potential reach. The difference in δ_p across platforms is neither statistically nor economically different. Developers face about the same challenges in reaching customers on both platforms. The similarity between δ_i and δ_d also means, when we take it together with the similarity of $(\alpha_{iX}, \beta_{iX})$ and $(\alpha_{dX}, \beta_{dX})$, that predicted demand for apps is symmetric across the two platforms.

Why do we find symmetry in the demand across platforms? Symmetry in the data seems to be the reason. If we look at Figure VI, we see that the distribution of observed reach (conditional on being over the comScore threshold on both platforms) is pretty close to symmetric empirically. As to the conditioning, in Figure I we see that the mean of reach for apps that are observed on only one platform is about the same for both platforms, and the probability of being observed in comScore on a platform conditional on being written for that platform is about the same for both platforms. The symmetry is in the data, not a modeling artifact.

VI.B. Supply for iOS vs. Android.

Developer incentives to write for each platform depend on expected realized reach, on profits per customer, on the size of the market, and on the fixed costs of writing for each platform.

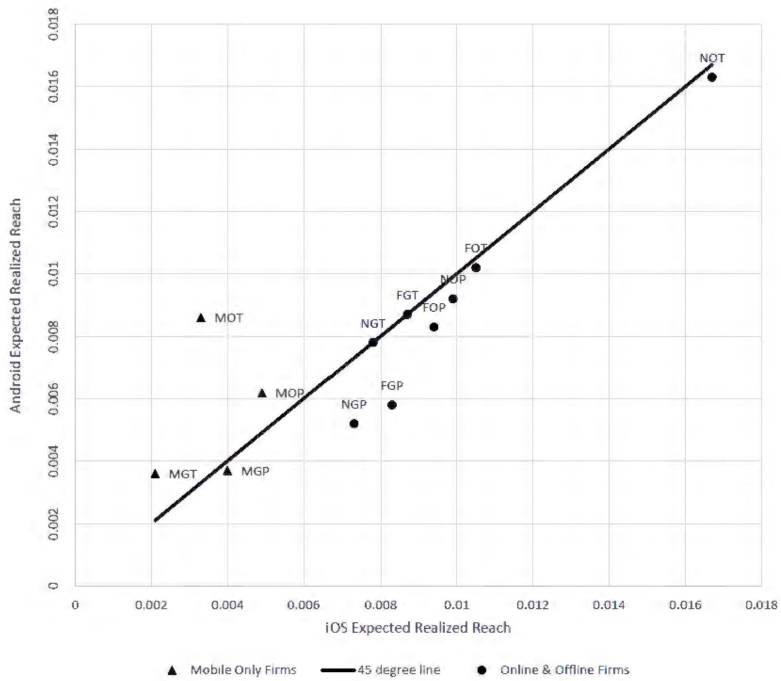
Three of these forces are estimated all together in our model: $\kappa_p = C_p/M_p/U_p$ so that κ_i and κ_d measure fixed costs normalized by per customer profit and the size of the market. We estimate, in an obvious notation, κ_{pX}, so that parameters vary by platform and by observable app type. We impose no restriction across platforms so that platform specific supply behavior could vary in ways not predicted by the demand parameters. There is also no statistical difference for each app type across platforms for κ_{pX}. This means that no evidence for observable variation in developer platform preferences.[13] We turn in a moment to the empirical foundation for this.

Much of developer incentives to supply an app is determined by expected realized demand. Indeed, the decision rule κ_p is simply a threshold for the $E[r_p]$ necessary to justify supplying an app to platform p. We can now use our parameter estimates to predict the expected realized reach of each type of app. Recall that nothing in our specification restricts $E[r_i|X]$ to be near $E[r_d|X]$ for the same conditioning X. These values are empirically identified by (1) the observed values r_p^* for a given X, when the app is supplied for platform p and goes over the 0.001 threshold in comScore, and (2) supply decisions S for a given X. In short, our estimates of the part of the correlation of r_i and r_d which are explained by variation in X is determined by the data and is not a modeling artifact.

Figure VII plots the joint distribution of $E[r_i|X]$ (on the horizontal axis) and $E[r_d|X]$ (on the vertical axis). Variation in X predicts substantial movement in demand on both platforms across observable app types, but the expected realized demands on the two platforms move closely together. The only type that falls far from the 45% line, mobile only, other (non-game), publicly-traded firms (MOT) has only one app in it. Bootstrapped tests of statistical significance confirm that there is no difference across platforms. Together with the very similar κ estimates, this implies highly symmetric supply. This finding, too, is largely data-driven. If we look at Figure I, we can see that developer supply decisions are highly symmetric: $S = (1,\ 1)$ is the most common choice, and $(0,\ 1)$ and $(1,\ 0)$ are approximately equally likely.

13. In our data, $U_i/U_d = .52/.4$, that is, there is a slightly higher installed base of Android phones than iPhones. We also have $\kappa_i \simeq \kappa_d$ which implies that $C_i/M_i/U_i$ is approximately equal to $C_d/M_d/U_d$. Rather than carrying around langauge that recognizes that we can't tell fixed costs from per-customer profits, we simply impose the not-identified conclusion that fixed costs are approximately equal across platforms and per-customer profits are slightly higher on iOS (.52/.4 higher.) This is completely consistent with what industry participants tell us and also saves carrying around a lot of langauge. We use the shorthand "similar fixed costs across platforms" for this.

FIGURE VII: JOINT EXPECTED REALIZED REACH ON iOS AND ANDROID BY APP-DEVELOPER TYPE



VI.C. Multihoming

Why is multihoming the most common supply behavior in our sample? We could summarize all the results discussed just above to provide a simple explanation of this. First, app demand and cost are close to symmetric across platforms at the current market shares. Second, at the app type level or at the individual app level, high expected demand on one platform predicts high expected demand on the other. As a result, developers with high expected reach apps on any platform have a strong incentive to multihome if the other platform has approximately the same number of users: the other platform represents half of their total potential profit. Third, the distribution of app demand, on either or both platforms, is highly skewed, so that a modest number of apps, highly popular on both platforms, tends to comprise much of demand and also to be inframarginal multihomers.

It is not clear that this result would extend to much less popular apps. Our sample selects developers based on the very low threshold that they gain one in a thousand potential customers on at least one platform. Thus we cannot learn much about the demand for, or behavior of, developers under that low threshold. For our purposes, this is not problematic. The contribution to app attractiveness of those apps too unpopular to be in our sample can safely be neglected.

Note that there is a pattern to the lower and higher reach apps and their types. In Figure VII, we observe that the app types with the lowest expected realized reach are by Mobile Only firms (MOT, MGT, MOP,

TABLE V: UNDERLYING MEANS AND COUNTS

	iOS $p = i$	Android $p = d$
$E[r_{pa} \vert a \in N_p]$	0.0100	0.0091
	(0.0059, 0.0122)	(0.0052, 0.0118)
$E[r_{pa} \vert a \in (N_{p'} \cap N_p)]$	0.0106	0.0102
	(0.0063, 0.0134)	(0.0070, 0.0134)

and MGP, designated by triangles). This may reflect demand preference difference between apps offered by entrepreneurial, Mobile Only firms and established firms (Online and Offline firms). This may also reflect differences faced by entrepreneurial and established firms in the marketing costs to reach customers. This finding corroborates our industry observations that the more established firms use existing customer relationships as a substitute for the marketing efforts required for entrepreneurial firms, and therefore could more easily reach more customers. The costs of the app store fall heavily on firms that are trying to acquire new customers in the mobile world (e.g. mobile entrepreneurs) and much less for those for whom the mobile world is an extension of existing customer relationships (e.g. banks). Regardless of the particular driver, the incentives to multihome are much greater for established firms due to higher realized reach than for entrepreneurial firms. These asymmetric reaches and thus incentives for multihoming indicate why established firms, rather than entrepreneurial firms, have played a much larger role in this industry than anticipated.

There is one conclusion from all these perspectives: mobile apps vary widely in their ability to reach consumers, whether due to preferences or costs of cutting through the app store, but the same mobile app has a strong tendency to have about the same reach to consumers on both platforms. The apps with the highest reach have the largest incentive to multihome, and these apps tend to be from established, not entrepreneurial, firms.

VI.D. Implications for Platform Tipping

Finally, we turn to the impact of these results for platform equilibrium. We saw above that developers of the most popular apps have strong incentives to mulithome. Here we examine the implications of that fact about developer behavior for the other side of the platform market, i.e. for users choosing a platform.

In Table V we report elements of an index number of the contribution of multihoming apps to all apps' attractiveness to users on the iOS and Android platforms. As we shall see, the contribution of multihoming apps is estimated to be substantially larger than the contribution of apps that choose only a single platform. Recall from our economic model of users that the index of the platform value to users due to app availability

is the sum of reaches of all apps available on that platform, $\sum_{a \in N_p} r_{pa}$. To calculate the portion of this which comes from multihoming apps, we simply split this sum up into two parts, the part which comes from multihoming apps and the part which comes from single-platform apps. This proportion is simply

(5) $$\frac{\sum_{a \in (N_{p'} \cap N_p)} r_{pa}}{\sum_{a \in N_p} r_{pa}} \times \frac{(N_{p'} \cap N_p)}{N_p}.$$

The first row of figures in Table V reports the mean realized reach for all apps that would have been written for each platform (i.e., $E[r_p] > \kappa_p$).[14] The second row of figures shows the reach for multihoming apps (i.e., $E[r_p] > \kappa_p \cap E[r_{p'}] > \kappa_{p'}$). Bootstrapped 95% confidence intervals are in parentheses below all figures. The percentage of multihoming apps relative to the total apps supplied for each platform (i.e., $(N_{p'} \cap N_p)/N_p$) are predicted to be 71.4% for Android and 71.0% for iOS. So for Android, the percent contribution of multihoming apps to the total value users derive from apps on Android is $0.0102/0.0091 \times 71.4\% = 80.0\%$. For iOS, the percent contribution of multihoming apps is $0.0106/0.0100 \times 71.0\% = 75.3\%$. The contribution of multihomers to platform value is slightly higher on Android than on iOS, but in both cases the majority of both platforms' values comes from multihomers.

These estimates are based on quantities observed in our sample data: the significantly higher mean reach for multihoming apps than for single platform apps. Looking back at Table V, we see that the multihoming apps tend to include the most popular apps, as measured by mean r^* and max r^*.

The combination of highly correlated and high reach on both platforms creates a strong incentive for the most valuable apps to multihome. As a result, consumers can access these applications, which comprise the bulk of value of apps, on either platform.

In results not reported in tables, we check the stability of our results over time by running our model on data from September, 2012. All results are similar. We interpret this as confirming that there has been little change in user preferences and developer incentives over the relevant time scale. Of course, checking over a much longer period of time might reveal movements in developer preferences.

14. We use population weights, rather than sample weights, to calculate this mean, since we are interested in the implications of our estimates for all developer activity (i.e., including those whose apps would not reach comScore's threshold on either platform). In the Appendix, we show the impact of correcting for sample selection.

VII. Conclusion

We have estimated a model of developer platform choice on a new dataset of mobile app developers' smartphone supply decisions. We examine the decision to supply for either or for both of the dominant platforms. Our model treats supplying for a platform much like models of entry treat serving a market: a platform will be supplied with an app only if expected demand for the app on that platform is high enough to generate variable profits that cover the incremental fixed costs of writing and marketing. Fortunately for identification, we have some outcome variables in addition to developers' app supply decisions. For apps that are at least moderately successful, we also observe quantity demanded in a sample of users.

Our model has a novel element that we hope is useful in entry studies generally. We show how to correct for the unavoidable problem that the set of potential entrants into markets is either unobserved or selected. In our application, the quantitative importance of selection is high, as we see large differences in estimated developer profitability between the selected sample of potential entrants and the much larger population.

We find that there are dramatic differences in app profitability by observable developer type and app type. From an economic growth perspective, the most important of these is the difference between entrepreneurial app developers and established firms. Since the mobile app platforms provide the vast bulk of the technical inputs needed to create a working system, many forecast, correctly, that there would be a wave of entrepreneurship in mobile apps. Many also forecast that this wave of entrepreneurs would disrupt existing businesses on a massive scale. Perhaps so someday, but at this stage the supply of mass market mobile apps is overwhelmingly from established firms. Based on our many industry interviews, we attribute much of the difference between firm types to marketing costs. It appears that the high costs of finding customers for new firms – some of which can be attributed to the considerable difficulty of matching buyer to seller on the platform-supplied app stores – are limiting the role of entrepreneurship.

The platform race on smartphones in the US has been approximately tied for a surprisingly long time. Our results provide an explanation of this otherwise quite surprising fact. As a threshold empirical result, we see that there is little difference in expected profitability for developers between the two platforms. This means that an equilibrium with divided platform choices exists. Such equilibria are typically not stable, and platform markets tend to tip from them. However, our other results explain why this market does not tip: (1) The distribution of app attractiveness to consumers is skewed, with a small minority of apps drawing the vast majority of consumer demand. (2) Apps which are highly demanded on one platform tend also to be highly demanded on the other platform. (3) These highly demanded apps have a strong tendency to multihome, writing for both platforms. As a result, the presence or absence of apps offers little reason for consumers to choose a platform. A consumer can choose either platform and have access to the most

attractive apps. This undercuts the mechanism by which the platform market might tip. Note that we do not address the sequential nature of entry on the platforms. While this is an interesting process to explore, for the purposes of this paper, our conclusions about the low likelihood of future market tipping are unaffected, since market shares of the platforms are already evenly split and the most popular have already multihomed. What would ordinarily be the unstable platform market equilibrium, with an approximate tie between iOS and Android, is rendered stable. This result, which we came to empirically, does not appear in the theoretical literature.

VIII. Bibliography

Augereau, Angelique, Shane Greenstein, and Marc Rysman. 2006. "Coordination vs. Differentiation in a Standards War: 56K Modems." The RAND Journal of Economics, 37(4): 771-88.

Bass, Dina and Adam Satariano - Jul 14, 2010. "Microsoft Pays Mobile App Developers to Help It Catch Apple", Bloomberg. http://www.bloomberg.com/news/print/2010-07-14/microsoft-pays-developers-to-build-mobile-apps-to-help-catch-up-with-apple.html accessed May 28, 2014.

Berry, S.T. (1992), "Estimation of a Model of Entry in the Airline Industry", Econometrica 60(4): 889-917.

Berry, Steven & Reiss, Peter, 2007."Empirical Models of Entry and Market Structure," in Handbook of Industrial Organization, vol. 3, Mark Armstrong and Robert Porter, eds. North-Holland Press

Berry, S.T., and E. Tamer (2006), "Identification in Models of Oligopoly Entry", xxxx

Berry, S.T., and J. Waldfogel (1999), "Social Inefficiency in Radio Broadcasting", Rand Journal of Economics, 30(3): 397-420.

Boudreau, K. (2010). Open Platform Strategies and Innovation: Granting Access vs. Devolving Control. Management Science. 56(10) 1849-1872.

Boudreau, K. (2012). Let a Thousand Flowers Bloom? An Early Look at Large Numbers of Software "Apps" Developers and Patterns of Innovation. Organization Science.

Boudreau, K. and A. Hagiu. 2009. Platforms Rules: Multi-sided Platforms as Regulators. A. Gawer, ed. Platforms, Markets and Innovation. Edward Elgar, London.

Bresnahan, Timothy F. and Shane Greenstein. 1999. Technological Competition and the Structure of the Computing Industry. Journal of Industrial Economics. 47(1): 1-40

Bresnahan, Timothy, and Shane Greenstein. 2014. "Mobile Computing: The Next Platform Rivalry." American Economic Review, 104(5): 475-80.

Bresnahan, T.F. and P.C. Reiss (1990), "Entry in Monopoly Markets", Review of Economic Studies 57:

57-81.

Bresnahan, T., Davis, J. P., & Yin, P. (2014) "Economic Value Creation in Mobile Applications." Forthcoming in NBER volume on Science and Innovation Policy, Adam Jaffe and Ben Jones, Editors.

Brown, J. and J. Morgan (2009) "How much is a Dollar Worth? Tipping versus Equilibrium Coexistence on Competing Online Auction Sites," Journal of Political Economy, 117: 668-700.

Cantillon, E. and P. Yin, "Competition Between Exchanges: Lessons from the Battle of the Bund," 2013

Casadesus-Masanell, Ramon, and Hanna Halaburda. "When Does a Platform Create Value by Limiting Choice?" Journal of Economics & Management Strategy 23, no. 2 (Summer 2014): 259–293.

Church, Jeffrey, and Neil Gandal. 1992. "Network Effects, Software Provision, and Standardization." Journal of Industrial Economics, 40(1): 85-103.

Ciliberto, F., and E. Tamer (2003), "Market Structure and Multiple Equilibria in the Airline Industry", xxx

Corts, Ken, and Mara Lederman. "Software Exclusivity and Indirect Network Effects in the US Home Video Game Industry." International Journal of Industrial Organization. 259-70.

Damiano, Ettore, and Hao Li. 2008. "Competing Matchmaking." Journal of the European Economic Association, 6(4):789-818.

Davis, J., Y. Muzyrya, and P. Yin, "Killer Apps," 2013.

Davis, P. (1997), "Spatial Competition in Retail Markets: Motion Theaters", Rand Journal of Economics.

Dranove, D., & Gandal, N. (2003). The Dvd-vs.-Divx Standard War: Empirical Evidence of Network Effects and Preannouncement Effects. Journal of Economics & Management Strategy, 12(3), 363-386.

Dubé, Jean-Pierre, K. Sudhir, Andrew Ching, Gregory S. Crawford, Michaela Draganska, Jeremy T. Fox, Wesley Hartmann, Günter J. Hitsch, V. Brian Viard, Miguel Villas-Boas "Recent Advances in Structural Econometric Modeling: Dynamics, Product Positioning and Entry" Marketing letters '05

Ellison, Glenn and Drew Fudenberg. 2003. "Knife-Edge or Plateau: When Do Market Models Tip?," Quarterly Journal of Economics, 118(4), 1249-1278.

Farrell, Joseph, and Paul Klemperer. "Coordination and lock-in: Competition with switching costs and network effects." Handbook of industrial organization 3 (2007): 1967-2072.

Greenstein, Shane, and Marc Rysman. 2007. "Coordination Costs and Standard Setting: Lessons from 56K Modems." In Standards and Public Policy, ed. S. Greenstein and V. Stango. Journal of the European Economic Association, 123-59. Cambridge University Press.

Lee, Robin. 2013. "Vertical Integration and Exclusivity in Platform and Two-Sided Markets."

Mazzeo, M. (2002), "Product Choice and Oligopoly Market Structure", Rand Journal of Economics 33(2): 1-22.

Rysman, M. (2004), "Competition Between Networks: A Study of the Market for Yellow Pages", Review of Economic Studies 71: 483-512.

Rysman, M. (2009). The economics of two-sided markets. The Journal of Economic Perspectives, 125-143.

Rysman, Marc. 2007. "An Empirical Analysis of Payment Card Usage." Journal of Industrial Economics, 55(1): 1-36.

Seim, K. (2006), An empirical model of firm entry with endogenous product-type choices. The RAND Journal of Economics, 37: 619–640

Sinkinson, Michael Pricing and Entry Incentives with Exclusive Contracts: Evidence from Smartphones xxx

Small, K. A. and H. S. Rosen. 1981. "Applied Welfare Economics with Discrete Choice Models," Econometrica, 49(1):105-130.

Sutton, J. 1991. Sunk Costs and Market Structure: Price Competition, Advertising, and the Evolution of Concentration. MIT Press.

Yin, Pai-Ling, Jason P. Davis, and Yulia Muzyrya. 2014. "Entrepreneurial Innovation: Killer Apps in the iPhone Ecosystem." American Economic Review, 104(5): 255-59.

Zhu, Ting, Hongju Liu and Pradeep K. Chintagunta. 2011. "Wireless Carriers' Exclusive Handset Arrangements: An Empirical Look at the iPhone," NET Institute

working paper.

IX.

Appendix

In Table X, we present the mean realized reach r_p for apps on Android and iOS based on our model. The first line of results shows these calculations using sample frequencies as weights. The second line of the results shows these calculations when we weight using the implied population frequencies calculated using Bayes law. Bootstrapped 95% confidence intervals are in parentheses below the calculations.

The difference between the sample and population results reflects the effect of accounting for entry selection. As expected, when we compare the first and second rows within each of the two columns, the sample reaches are higher than the population reaches, since the sample does not consider the distribution of regressors in the apps which did not reach the comScore observation threshold. Those would have been apps with lower reach, and thus they bring down the mean reach in the population estimates. Bootstrapped

TABLE VI: MODEL PARAMETER ESTIMATES (GAMES, PRIVATE)

Parameter	Android	iOS
τ	.87	
	(0.39,1.00)	
δ	.19	.21
	(0.11,0.96)	(0.12,0.90)
Online Firm		
α	1.46	0.94
	(0.22,2.33)	(0.20,1.86)
β	62.2	31.8
	(28.0,81.5)	(17.9,61.5)
κ	.0054	.0072
	(.0026,.0085)	(.0036,.0128)
ω	.56	
	(.10,.96)	
Mobile Only Firm		
α	1.08	0.69
	(0.08,1.54)	(0.07,1.79)
β	62.4	41.8
	(31.1,83.2)	(29.1,73.9)
κ	.0033	.0035
	(.0006,.0040)	(.0008,.0052)
ω	.11	
	(.20,.68)	
Offline Firm		
α	2.28	2.62
	(0.04,2.47)	(0.14,3.70)
β	86.4	75.6
	(30.8,102.1)	(28.4,101.0)
κ	.0052	.0074
	(.0006,.0071)	(.0014,.0110)
ω	.61	
	(.26,.91)	

TABLE VII: MODEL PARAMETER ESTIMATES (GAMES, PUBLICLY TRADED)

τ	.87	
	(0.39,1.00)	
δ	.19	.21
	(0.11,0.96)	(0.12,0.90)
Online Firm		
α	1.14	0.46
	(0.12,4.48)	(0.11,2.26)
β	45.4	14.9
	(24.7,66.2)	(6.50,50.2)
κ	.0091	.0102
	(.0046,.0210)	(.0062,.0198)
ω	.88	
	(.60,1.0)	
Mobile Only Firm		
α	0.76	0.20
	(0.04,2.83)	(0.03,1.54)
β	45.6	24.9
	(29.5,76.4)	(14.2,62.2)
κ	.0069	.0065
	(.00153,.0184)	(.0022,.0171)
ω	.43	
	(.35,.96)	
Offline Firm		
α	1.96	2.13
	(0.30,3.17)	(0.33,3.15)
β	69.6	58.7
	(33.2,83.9)	(16.9,80.3)
κ	.0089	.0103
	(.0038,.0223)	(.0056,.0209)
ω	.93	
	(.69,.97)	

TABLE VIII: MODEL PARAMETER ESTIMATES (OTHER, PRIVATE)

τ	.87	
	(0.39,1.00)	
δ	.19	.21
	(0.11,0.96)	(0.12,0.90)
Parameter	Android	iOS
Online Firm		
α	1.27	0.96
	(0.22,2.81)	(0.23,2.55)
β	33.3	25.8
	(18.7,49.6)	(13.4,45.4)
κ	.0072	.0079
	(.0050,.0122)	(.0045,.0133)
ω	.49	
	(.00,.87)	
Mobile Only Firm		
α	0.89	0.70
	(0.16,1.96)	(0.15,1.79)
β	33.4	35.8
	(12.6,65.4)	(26.3,63.5)
κ	.0050	.0042
	(.0030,.0092)	(.0025,.0064)
ω	.05	
	(.00,.64)	
Offline Firm		
α	2.09	2.64
	(0.28,2.36)	(0.34,3.97)
β	57.4	69.6
	(23.9,70.9)	(26.9,80.0)
κ	.0070	.0081
	(.0037,.0111)	(.0044,.0112)
ω	.54	
	(.17,.89)	

TABLE IX: MODEL PARAMETER ESTIMATES (OTHER, PUBLICLY TRADED)

τ	.87	
	(0.39,1.00)	
δ	.19	.21
	(0.11,0.96)	(0.12,0.90)
Parameter	Android	iOS
Online Firm		
α	0.95	0.47
	(0.10,4.97)	(0.08,3.20)
β	16.5	8.95
	(6.21,45.3)	(4.42,41.1)
κ	.0109	.0109
	(.0081,.0226)	(.0075,.0215)
ω	.81	
	(.49,.96)	
Mobile Only Firm		
α	0.57	0.22
	(0.05,3.96)	(0.03,2.42)
β	16.6	19.0
	(3.55,55.2)	(6.69,52.2)
κ	.0087	.0072
	(.0043,.0200)	(.0030,.0188)
ω	.37	
	(.10,.89)	
Offline Firm		
α	1.77	2.15
	(0.30,3.72)	(0.37,3.30)
β	40.6	52.7
	(22.3,50.1)	(19.5,57.2)
κ	.0107	.0111
	(.0074,.0239)	(.0076,.0226)
ω	.86	
	(.29,.95)	

TABLE X: Model Predictions of Reach Moments

	Mean Predicted Reach on Android Realized	Mean Predicted Reach on iOS Realized
In Sample	0.0076	0.0079
	(0.0054,0.0105)	(0.0062,0.0103)
In Population	0.0070	0.0076
	(0.0036,0.0089)	(0.0033,0.0089)

95% confidence intervals for the differences between population and sample means are statistically signficant for both Android and iOS.